

# Zadanie Banki

Limit czasu: 2 s, limit pamięci: 256 MB

Oryginał zadania pochodzi z platformy Codeforces. [\[link\]](#)

---

## Treść

Marek kocha pieniądze i dla bezpieczeństwa większość z nich trzyma w bankach. Dokładniej, trzyma je w  $n$  różnych bankach, a w  $i$ -tym banku znajduje się  $a_i$  euro.

Pewnego dnia Marek postanowił zgromadzić wszystkie swoje pieniądze w jednym banku. W tym celu będzie musiał dokonać wiele przelewów między swoimi kontami w różnych bankach. Wszystkie przelewy międzybankowe odbywają się w ten sam sposób: w ramach jednego przelewu może przelać tylko dokładnie  $x$  euro, a uwzględniając wszystkie opłaty, na drugie konto zostanie przelane  $y$  euro (ponieważ banki chcą zarobić, zachodzi zależność  $y \leq x$ ). Nie ma ustalonego limitu liczby przelewów w żadnym z banków.

Może się zdarzyć, że Marek nie będzie w stanie przelać wszystkich swoich pieniędzy do jednego banku, ale chce znaleźć maksymalną liczbę euro, jaka może ostatecznie znaleźć się w dowolnym (jednym) banku.

## Wejście

Każdy test zawiera wiele zestawów danych. Pierwsza linia wejścia zawiera  $t$  ( $1 \leq t \leq 10^4$ ) – liczbę zestawów danych które pojawią się na wejściu.

Każdy zestaw danych składa się z dwóch linii:

- Pierwsza zawiera trzy liczby całkowite –  $n$ ,  $x$  oraz  $y$  ( $2 \leq n \leq 2 \cdot 10^5$ ,  $1 \leq y \leq x \leq 10^9$ ) – odpowiednio ilość banków, kwota przelewu oraz kwota, która dociera do adresata.
- Druga zawiera  $n$  liczb całkowitych  $a_1, a_2, \dots, a_n$  ( $1 \leq a_i \leq 10^9$ ), gdzie  $a_i$  początkowy stan konta w  $i$ -tym banku.

Gwarantowane jest, że suma  $n$  dla wszystkich zestawów danych w teście nie przekracza  $2 \cdot 10^5$ .

## Wyjście

Dla każdego przypadku testowego wypisz jedną liczbę – maksymalną liczbę euro, jaką można zgromadzić w dowolnym banku.

# Zadanie Banki

Limit czasu: 2 s, limit pamięci: 256 MB

Oryginał zadania pochodzi z platformy Codeforces. [\[link\]](#)

---

## Przykłady

Wejście (przykład 1):

```
6
4 5 4
10 9 8 7
5 13 11
47 52 64 13 91
2 1 1
1000 1000
3 15 14
34 43 52
6 7 6
15 17 14 15 12 16
2 15 10
45 44
```

Wyjście (przykład 1):

```
25
229
2000
113
72
74
```

### Wyjaśnienie:

W pierwszym przypadku, optymalna sekwencja przelewów międzybankowych wygląda tak:

$$1 \rightarrow 4, 1 \rightarrow 4, 4 \rightarrow 3, 4 \rightarrow 3, 4 \rightarrow 3, 3 \rightarrow 2, 3 \rightarrow 2, 3 \rightarrow 2, 3 \rightarrow 2.$$

Tak wyglądają zmiany po każdym przelewie:

$$(10, 9, 8, 7)$$
$$\xrightarrow{1 \rightarrow 4} (5, 9, 8, 11) \xrightarrow{1 \rightarrow 4} (0, 9, 8, 15)$$
$$\xrightarrow{4 \rightarrow 3} (0, 9, 12, 10) \xrightarrow{4 \rightarrow 3} (0, 9, 16, 5) \xrightarrow{4 \rightarrow 3} (0, 9, 20, 0)$$
$$\xrightarrow{3 \rightarrow 2} (0, 13, 15, 0) \xrightarrow{3 \rightarrow 2} (0, 17, 10, 0) \xrightarrow{3 \rightarrow 2} (0, 21, 5, 0) \xrightarrow{3 \rightarrow 2} (0, 25, 0, 0).$$

W rezultacie jest możliwe otrzymać 25 euro w jednym banku (dokładniej w drugim) i to właśnie odpowiedź do tego zestawu danych.

W trzecim przypadku, możliwe jest przelanie jednego euro z pierwszego do drugiego banku 1000 razy i otrzymać 2000 euro (w drugim banku).